Disentangling unwanted sources of variation in single-cell RNA-sequencing data under weak supervision

Hui Ting Grace Yeo
Massachusetts Institute of Technology
Cambridge, MA
yhtgrace@mit.edu

David K Gifford
Massachusetts Institute of Technology
Cambridge, MA
gifford@mit.edu

Abstract

The presence of unwanted sources of variation presents a major challenge for the analysis of large multiplexed perturbational single-cell RNA sequencing (scRNA-seq) studies. Removal of these nuisance factors typically requires expert knowledge to identify the factors and the tedious curation of factor associated gene sets. We propose instead to model unwanted factors with a deep generative modeling framework under the weak supervision of a control population. Gene expression of both control and treatment populations are jointly modeled as being generated from two sets of disentangled latent variables. One variable corresponds to variation found in both datasets, while the other variable is constrained to vary only to explain the treatment dataset. Applying our model to a perturbational dataset where cell cycle is confounding, we find that our model is not only able to learn the shared source of variation *de novo* without expert annotation but also learns a more disentangled representation of the perturbational effects than linear baselines.

1 Introduction

Single-cell RNA-sequencing is increasingly being used in large multiplexed experiments to read out gene expression changes in response to various perturbations [1–3]. One challenge is that the starting population of cells is often already heterogeneous, hence complicating downstream analysis of the perturbational effects of interest. Current approaches to this problem utilize expert knowledge to identify these confounding factors [4, 5]. However, these factors are not always easily identifiable, and their removal may require painstaking curation of the associated gene set.

Deep generative models have been shown to learn disentangled latent spaces without any supervision [6–8]. More recently, weakly supervised models that take advantage of auxiliary data have also been described. In particular, a recent work by Ruiz et al. [9] describes a method that uses a reference dataset in which factors of interest are constant to improve disentanglement. Inspired by this, we hypothesized that in a perturbational setting a control population can be used as a source of weak supervision to disentangle non-treatment related

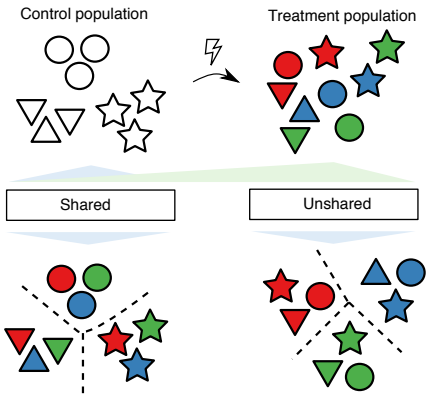

Figure 1: **Problem setting:** Control and treatment populations from a perturbational experiment are jointly modeled as being generated from two sets of disentangled latent variables. Shapes correspond to variation present in the control population, while colors correspond to variation introduced by perturbations.

14th Machine Learning in Computational Biology (MLCB 2019), Vancouver, Canada.

from treatment related variation. We jointly model the gene expression data of the control and treatment populations as being generated from two sets of disentangled latent variables: shared and unshared. Shared latent variables capture variation present in the control and treatment datasets while the unshared latent variables are constrained to vary only for the treatment dataset.

We fit our model to perturbational data in which it was previously observed that cell cycle effects were confounding in the treatment population. To evaluate our model, we train low capacity models to predict treatment and cell cycle effects from each set of latent variables. We show that our model is not only able to learn the shared source of variation without expert knowledge or annotation but also learns a more disentangled representation of the perturbational effects than our baselines. We expect this approach to be broadly useful as these representations are straightforward to use in common downstream analyses such as clustering and pseudo-time analysis.

2 Related Work

General approaches for removing unwanted sources of variation in scRNA-seq data are limited because they typically rely upon expert annotation of confounders, or do not utilize data from control experiments. For example, both f-scLVM and PAGODA make use of annotations derived from public databases such as MSigDB and REACTOME [4, 5] to identify confounding factors and their associated gene sets. Similarly, commonly-used single-cell preprocessing pipelines such as Seurat and SimpleSingleCell use pre-trained classifiers based on curated gene sets to identify cell cycle effects [10, 11]. To identify unannotated factors, f-scLVM makes sparsity assumptions about their effects rather than using an external control dataset. In the analysis of their perturbational datasets, Adamson et al. [1] and Dixit et al. [3] describe PCA-based models for estimating variation attributable to unwanted factors. However, these models are linear and hence limited in their expressiveness.

In representation learning, deep generative models, such as the variational autoencoder, describe joint distributions over the data and a latent code [6]. In particular, there has been great interest in discovering interpretable representations that are disentangled with respect to the actual generative factors– that is, a given latent variable should vary according to a single generative factor, but be invariant with respect to all others. Some examples of prominent work include β-VAE and infoGAN [8, 7, 12]. While deep generative models are usually unsupervised, there has also been recent work in the semi-supervised and weakly supervised settings that exploit partial annotation or auxiliary data sources [13, 14]. In particular, we are inspired by the reference-based VAE (rb-VAE) model described by Ruiz et al. [9] that makes use of a reference dataset in which factors of interest are constant is available. In this work, we adapt this framework for use in the analysis of perturbational scRNA-seq datasets.

3 Methods

Model and task description: Let us consider gene expression observations $\mathbf{x} \in \mathbb{R}^m$ of m genes from two cell populations: $\mathbf{T} = \{\mathbf{x}_t^i\}_{i=1}^{N_t}$, which is the set of N_t cells sampled from the treatment population, and $\mathbf{C} = \{\mathbf{x}_c^i\}_{i=1}^{N_c}$, which is the set of N_c cells sampled from the control population. We jointly model the datasets as being generated by a random process given two sets of generative factors $\mathbf{s} \in \mathbb{R}^{k_s}$ and $\mathbf{u} \in \mathbb{R}^{k_u}$, that we respectively refer to as *shared latent factors* and *unshared latent factors*. In general, $\mathbf{k}_s, \mathbf{k}_u << m$.

Our goal is to learn a disentangled representation such that $\mathbf{s}$ captures variation present in both datasets, while $\mathbf{u}$ captures variation present only in the treatment dataset. Note that this implies that $\mathbf{u}$ should be constant for the control dataset. Hence, we can state the generative process for each population as follows: for the treatment population, $\mathbf{s}^i$ and $\mathbf{u}^i$ are first drawn from their respective prior distributions $p(\mathbf{s})$ and $p(\mathbf{u})$. Then, $\mathbf{x}_t^i$ is drawn from the conditional distribution $p_\theta(\mathbf{x}|\mathbf{s}, \mathbf{u})$. In contrast, for the control population, $\mathbf{s}^i$ is still drawn per data point, but only a single draw is made for $\mathbf{u}$. $\mathbf{x}_c^i$ is then drawn from the conditional distribution $p_\theta(\mathbf{x}|\mathbf{s}, \mathbf{u} = \hat{\mathbf{u}}_c)$. Formally, our task is to perform approximate inference of the latent variables $\mathbf{s}$ and $\mathbf{u}$.

Inference: We perform inference on the model following Ruiz et al. [9]. For the treatment dataset, we choose the prior over the latent variables to be $p(\mathbf{s}) = \mathcal{N}(0, \mathbf{I})$ and $p(\mathbf{u}) = \mathcal{N}(0, \mathbf{I})$. For the control dataset, the prior over $\mathbf{s}$ is similarly chosen to be $\mathcal{N}(0, \mathbf{I})$ but the prior over $\mathbf{u}$ is chosen to be

flat. Since the output is real-valued, we let the generator $p_\theta(\mathbf{x}|\mathbf{s},\mathbf{u})$ be multivariate Gaussian with the mean given by a neural net with parameters θ.

The true posterior of the generative model i.e. $p_\theta(\mathbf{s},\mathbf{u}|\mathbf{x})$, is intractable, and hence approximated by a factorizable probabilistic encoder $q_\phi(\mathbf{s},\mathbf{u}|\mathbf{x}) = q_\phi(\mathbf{s}|\mathbf{x})q_\phi(\mathbf{u}|\mathbf{x})$. We define $q_\phi(\mathbf{s}|\mathbf{x^{(i)}}) = \mathcal{N}(\mathbf{s};\mu^{(i)},\sigma^{(i)2})$ to be Gaussian with mean and standard deviation also computed by a neural net with parameters ϕ. $q_\phi(\mathbf{u}|\mathbf{x}^{(i)})$ is similarly defined. The reparameterization trick then allows for optimization of the following objective via stochastic gradient descent:

$$\min_{\theta,\phi,\hat{\mathbf{u}}_c} \mathbb{E}_{\mathbf{x}\sim\mathbf{T}}[\mathbb{KL}(q_\phi(\mathbf{s}|\mathbf{x})q_\phi(\mathbf{u}|\mathbf{x})||p(\mathbf{s})p(\mathbf{u})) - \mathbb{E}_{q_\phi(s|x)q_\phi(u|x)}\log p_\theta(\mathbf{x}|\mathbf{s},\mathbf{u})]+$$
$$\mathbb{E}_{\mathbf{x}\sim\mathbf{C}}[\mathbb{KL}(q_\phi(\mathbf{s}|\mathbf{x})||p(\mathbf{s})) - \mathbb{E}_{q_\phi(s|x)}\log p_\theta(\mathbf{x}|\mathbf{s},\hat{\mathbf{u}}_\mathbf{c})]$$

One problem is that since samples from the control population are reconstructed directly from the learned parameter $\hat{\mathbf{u}}_\mathbf{c}$, $q_\phi(\mathbf{u}|\mathbf{x})$ does not necessarily have to learn to encode $\mathbf{u}_c$ given $\mathbf{C}$. That is, $q_\phi(\mathbf{u}|\mathbf{x})$ could learn to encode samples from $\mathbf{C}$ arbitrarily, since samples are not reconstructed from doing inference on $\mathbf{u}_c$. However, it should be desirable for $q_\phi(\mathbf{u}|\mathbf{x})$ to also encode $\mathbf{u}_c$ in a semantically meaningful way with respect to $\mathbf{u}_t$. Hence, we propose to also maximize the likelihood of encoding samples of $\mathbf{C}$ as $\hat{\mathbf{u}}_\mathbf{c}$ i.e. by also minimizing $\mathbb{E}_{\mathbf{x}\sim\mathbf{C}}\log q_\phi(\hat{\mathbf{u}}_\mathbf{c}|\mathbf{x})$. We refer to the model fit with the original framework as rb-VAE, and this modification as rb-VAE-e. Further implementation details may be found in the appendix.

4 Results

Dataset: We consider a perturbational single-cell RNA-sequencing dataset in which cell cycle was observed to be a major source of confounding variation. In this dataset, all single, double and triple combinations of CRISPR perturbations targeting the three branches of mammalian unfolded protein response (UPR) were introduced to K562 cells. Then, cells were treated with a pharmacological inducer of UPR, thapsigargin. In addition, a control cell population was treated with DMSO [1]. We hypothesized that cell cycle should be a prominent source of variation in the control population, and so the control population would provide good supervision for our modeling framework. The datasets were preprocessed using Seurat's standard preprocessing pipeline (See appendix) [10]. The model was fit using 10 random 80-20 training-test splits of the data.

Baseline models: We compared our model to two PCA-based linear baselines: In the first, PCA was first fit to the control dataset. A second PCA was then fit to the residual, which was computed by subtracting the treatment dataset from its reconstruction using the first PCA. In the second, after computing the scores using the first PCA on the treatment dataset, we fit a linear regression model to predict the treatment data from its scores instead. A second PCA is then fit to the residual of the linear model. We refer to this second approach as PCA-lr. In both cases, the scores of the first PCA were regarded as the shared variables, while the scores of the second PCA were regarded as the unshared variables. We also compared our model to the fully unsupervised method β-VAE. Since β-VAE does not automatically partition its variables into shared and unshared, we defined them as the top variables most predictive of each source of variation. For all models, we set the size of the unshared and shared latent variables to 8, for a total of 16.

Reconstruction error: We first evaluated reconstruction error on the held-out test set. As expected, we observe that the original rb-VAE framework can result in arbitrary decoding of the control dataset, as can be seen from the higher reconstruction error when performing inference on $\mathbf{u}$. Our modified framework achieves lower reconstruction error on not only the control but also the treatment dataset, suggesting that the model is encoding $\mathbf{u_c}$ in a more meaningful way with respect to $\mathbf{u_t}$.

Disentanglement metrics: To measure disentanglement, we fit low capacity linear models to predict the known factors of variation given $\mathbf{s}$, $\mathbf{u}$, and the concatenated set $[\mathbf{s};\mathbf{u}]$, using 80-20 training-test splits. We fit logistic regression models to predict perturbation identity, and elastic net models to predict the expression of three cell cycle genes that are highly variable in the dataset: CCNB1, CENPA, and PKL1 (See appendix). In all cases, 5-fold cross-validation was performed over the training split to select regularization strength.

Table 1: **Model benchmarking results:** For reconstruction, we report mean squared error averaged over ten random held-out test sets. For disentanglement, we report macro-averaged F1 performance on predicting the perturbation received and r2 on predicting expression of CCNB1 on a randomly held-out test set. Mean and standard deviation is reported over the ten random training/test splits. Top two best methods for each metric is highlighted in bold. Full results can be found in the appendix.

	Reconstruction		Disentanglement			
			s		**u**	
Model	Control	Treatment	F1	r2 (CCNB1)	F1	r2 (CCNB1)
rb-VAE	3705.51	1918.45	**.183 ± .033**	.383 ± .019	.739 ± .014	.106 ± .009
rb-VAE-e	1722.06	1905.80	**.214 ± .020**	.403 ± .016	**.760 ± .019**	**.095 ± .008**
PCA	1730.72	1908.29	.242 ± .009	**.449 ± .005**	**.762 ± .006**	.174 ± .004
PCA-lr	1726.22	1915.29	.242 ± .009	**.449 ± .005**	.615 ± .013	**.022 ± .008**
β-VAE, β=1	1754.85	1913.89	.544 ± .123	.432 ± .031	.709 ± .023	.264 ± .083
β-VAE, β=2	1728.14	1888.52	.544 ± .074	.439 ± .019	.678 ± .015	.263 ± .066
β-VAE, β=4	1713.67	1918.88	.510 ± .053	.411 ± .028	.634 ± .018	.265 ± .055

We find that the unshared variables learned by rb-VAE-e are highly predictive of perturbation identity relative to predictive performance on the concatenated set, as well as other methods. Furthermore, it is less correlated with cell cycle effects (Table 1, Appendix). For example, the unshared variables learned by PCA are also strongly predictive of perturbation identity but are more correlated with cell cycle effects. We visualize this in Figure 2 by computing the UMAP projection of the latent spaces. Similarly, the unshared variables learned by PCA-lr are very weakly predictive of cell cycle effects, but are far less predictive of perturbation identity.

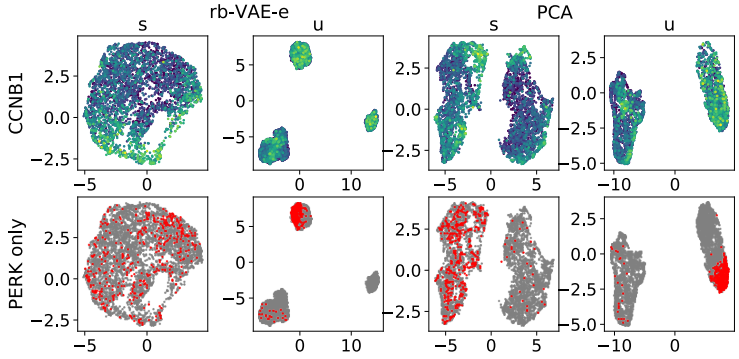

Figure 2: **UMAP visualization of latent variables learned by rb-VAE-e and PCA** for a particular seed. Top: cells are colored by CCNB1 expression. Bottom: cells are colored by whether they received a gRNA targeting only PERK

We also observe that the shared variables learned by our model are both less predictive of perturbation identity and cell cycle effects than other models. However, we note that the concatenated set for rb-VAE-e is also slightly less predictive of cell cycle (See appendix). We hypothesize that given the slightly lower reconstruction error of our model, it could be that $\mathbf{s}$ is encoding more variation within the dataset and hence presents a harder task of predicting cell cycle effects. Model performance on the other two cell cycle genes tested followed the same trends as CCNB1.

Both rb-VAE variants and the linear baselines outperformed the fully unsupervised β-VAE, suggesting that methods that make explicit use of the control population are better at disentanglement.

5 Conclusion

In this work, we present a modeling framework for disentangling unwanted sources of variation in a perturbational setting by exploiting information available in a control dataset. We show that our model not only learns the unwanted source of variation without any annotation required but also produces a more disentangled representation of the perturbational effects than unsupervised approaches as well as weakly supervised linear baselines. Representations learned by the model can be used in typical downstream analysis tasks that require low-dimensional projections of scRNA-seq data.

References

[1] Britt Adamson, Thomas M Norman, Marco Jost, Min Y Cho, James K Nunez, Yuwen Chen, Jacqueline E Villalta, Luke A Gilbert, Max A Horlbeck, Marco Y Hein, Ryan A Pak, Andrew N Gray, Carol A Gross, Atray Dixit, Oren Parnas, Aviv Regev, and Jonathan S Weissman. A multiplexed Single-Cell CRISPR screening platform enables systematic dissection of the unfolded protein response. *Cell*, 167(7):1867–1882.e21, December 2016.

[2] Paul Datlinger, André F Rendeiro, Christian Schmidl, Thomas Krausgruber, Peter Traxler, Johanna Klughammer, Linda C Schuster, Amelie Kuchler, Donat Alpar, and Christoph Bock. Pooled CRISPR screening with single-cell transcriptome readout. *Nat. Methods*, 14(3):297–301, March 2017.

[3] Atray Dixit, Oren Parnas, Biyu Li, Jenny Chen, Charles P Fulco, Livnat Jerby-Arnon, Nemanja D Marjanovic, Danielle Dionne, Tyler Burks, Raktima Raychowdhury, Britt Adamson, Thomas M Norman, Eric S Lander, Jonathan S Weissman, Nir Friedman, and Aviv Regev. Perturb-Seq: Dissecting molecular circuits with scalable Single-Cell RNA profiling of pooled genetic screens. *Cell*, 167(7):1853–1866.e17, December 2016.

[4] Florian Buettner, Naruemon Pratanwanich, Davis J McCarthy, John C Marioni, and Oliver Stegle. f-scLVM: scalable and versatile factor analysis for single-cell RNA-seq. *Genome Biol.*, 18(1):212, November 2017.

[5] Jean Fan, Neeraj Salathia, Rui Liu, Gwendolyn E Kaeser, Yun C Yung, Joseph L Herman, Fiona Kaper, Jian-Bing Fan, Kun Zhang, Jerold Chun, and Peter V Kharchenko. Characterizing transcriptional heterogeneity through pathway and gene set overdispersion analysis. *Nat. Methods*, 13(3):241–244, March 2016.

[6] Diederik P Kingma and Max Welling. Auto-encoding variational bayes, 2013.

[7] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-VAE: Learning basic visual concepts with a constrained variational framework. November 2016.

[8] Christopher P Burgess, Irina Higgins, Arka Pal, Loic Matthey, Nick Watters, Guillaume Desjardins, and Alexander Lerchner. Understanding disentangling in β-VAE. April 2018.

[9] Adria Ruiz, Oriol Martinez, Xavier Binefa, and Jakob Verbeek. Learning disentangled representations with Reference-Based variational autoencoders. January 2019.

[10] Andrew Butler, Paul Hoffman, Peter Smibert, Efthymia Papalexi, and Rahul Satija. Integrating single-cell transcriptomic data across different conditions, technologies, and species. *Nat. Biotechnol.*, 36(5):411–420, June 2018.

[11] Aaron T L Lun, Davis J McCarthy, and John C Marioni. A step-by-step workflow for low-level analysis of single-cell RNA-seq data with bioconductor. *F1000Res.*, 5:2122, August 2016.

[12] Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. InfoGAN: Interpretable representation learning by information maximizing generative adversarial nets. June 2016.

[13] Diederik P Kingma, Danilo J Rezende, Shakir Mohamed, and Max Welling. Semi-Supervised learning with deep generative models. June 2014.

[14] N Siddharth, Brooks Paige, Jan-Willem van de Meent, Alban Desmaison, Noah D Goodman, Pushmeet Kohli, Frank Wood, and Philip H S Torr. Learning disentangled representations with Semi-Supervised deep generative models. June 2017.

Appendix

1 Implementation

The architecture of the neural networks used for the deep generative models (rb-VAE, rb-VAE-e, β-VAE) are as follows:

```
Encoder(
  (encoder): Sequential(
    (0): Linear(in_features=2159, out_features=1024, bias=True)
    (1): LeakyReLU(negative_slope=True)
    (2): Linear(in_features=1024, out_features=1024, bias=True)
    (3): LeakyReLU(negative_slope=True)
    (4): Linear(in_features=1024, out_features=16, bias=True)
  )
)

Decoder(
  (decoder): Sequential(
    (0): Linear(in_features=16, out_features=1024, bias=True)
    (1): LeakyReLU(negative_slope=True)
    (2): Linear(in_features=1024, out_features=1024, bias=True)
    (3): LeakyReLU(negative_slope=True)
    (4): Linear(in_features=1024, out_features=2159, bias=True)
  )
)
```

All models were fit using the adam optimizer with a learning rate of 1e-4, $\beta = (0.9, 0.999)$ and a batch size of 64. Models were trained for a maximum of 250 epochs. The model with the best error on the training set was then used for downstream evaluation. All neural network models were implemented in pytorch 1.1.0.

Regarding inference for rb-VAE/rb-VAE-e: Ruiz et al. [1] describe in their work a symmetric variational extension to prevent the model from finding the solution $p(\mathbf{x}|\mathbf{s}, \mathbf{u}) = p(\mathbf{x}|\mathbf{s})$ in the case where $\mathbf{s}$ is too expressive. Empirically, we observe that limiting the size of $\mathbf{s}$ prevents the generator from ignoring the unshared variable and allows for a simpler learning algorithm.

Logistic regression and elastic net models are default models implemented in scikit-learn 0.21.2. They were trained using 5-fold cross-validation on the training set to select regularization strength.

2 Data preprocessing

Raw data was downloaded from GEO at accession number GSM2406677. Cells flagged as having good coverage, containing only 1 cell, and with a assigned guide identity were retained. Data was then processed using Seurat's default pipeline [2]. Thapsigargin-treated cells (gem group 2) were assigned as treatment cells, while DMSO-treated cells (gem group 3) were assigned as control cells. A set of differentially expressed genes with respect to the perturbations were determined as described by the original authors (i.e. mean UMIS per cell $\geq$ 0.5 and KS test statistic D 0.15 for at least one

Preprint. Under review.

perturbation) [3]. This resulted in a set of 4722 cells and 2159 genes. Data was then scaled for input into downstream models.

3 Cell cycle genes

The cell cycle genes chosen for estimating cell cycle effects are three highly variable genes within the gene set previously known to be associated with cell cycle. CCNB1 encodes for G2/mitotic-specific Cyclin B1 and is expressed in the G2/M phase. [4]. CENPA encodes for CENP-A is regulated by cell cycle and is involved in centromere function [5]. PLK1, also known as STPK13, is also known to vary with cell cycle, and is most highly expressed in G2/M [6].

4 Extended results

4.1 Full benchmarking results

Table 1: **Model benchmarking results on s**

	Perturbation identity			Cell cycle effects		
Model	AUROC	AUPRC	F1	r2 (CCNB1)	r2 (CENPA)	r2 (PLK1)
rb-VAE	.221 ± .033	.210 ± .030	.183 ± .033	.383 ± .019	.468 ± .010	.421 ± .016
rb-VAE-e	.245 ± .022	.244 ± .021	.214 ± .020	.403 ± .016	.473 ± .006	.435 ± .009
PCA	.288 ± .012	.275 ± .007	.242 ± .009	.449 ± .005	.494 ± .002	.473 ± .004
PCA-lr	.288 ± .012	.275 ± .007	.242 ± .009	.449 ± .005	.494 ± .002	.473 ± .004
β-VAE, β=1	.556 ± .106	.551 ± .118	.544 ± .123	.432 ± .031	.516 ± .016	.493 ± .013
β-VAE, β=2	.553 ± .071	.550 ± .071	.544 ± .074	.439 ± .019	.496 ± .018	.481 ± .016
β-VAE, β=4	.520 ± .047	.517 ± .054	.510 ± .053	.411 ± .028	.465 ± .024	.435 ± .032

Table 2: **Model benchmarking results on u**

	Perturbation identity			Cell cycle effects		
Model	AUROC	AUPRC	F1	r2 (CCNB1)	r2 (CENPA)	r2 (PLK1)
rb-VAE	.749 ± .014	.733 ± .014	.739 ± .014	.106 ± .009	.117 ± .023	.156 ± .024
rb-VAE-e	.771 ± .020	.754 ± .019	.760 ± .019	.095 ± .008	.097 ± .016	.130 ± .014
PCA	.771 ± .005	.756 ± .007	.762 ± .006	.174 ± .004	.304 ± .012	.269 ± .008
PCA-lr	.635 ± .013	.611 ± .012	.615 ± .013	.022 ± .008	.044 ± .002	.044 ± .003
β-VAE, β=1	.720 ± .021	.703 ± .024	.709 ± .023	.264 ± .083	.276 ± .104	.313 ± .112
β-VAE, β=2	.685 ± .014	.674 ± .015	.678 ± .015	.263 ± .066	.261 ± .091	.302 ± .095
β-VAE, β=4	.643 ± .018	.632 ± .018	.634 ± .018	.265 ± .055	.250 ± .057	.308 ± .075

Table 3: **Model benchmarking results on** $[\mathbf{s}; \mathbf{u}]$

	Perturbation identity			Cell cycle effects		
Model	AUROC	AUPRC	F1	r2 (CCNB1)	r2 (CENPA)	r2 (PLK1)
rb-VAE	.771 ± .013	.758 ± .013	.763 ± .013	.472 ± .009	.539 ± .007	.510 ± .009
rb-VAE-e	.788 ± .017	.773 ± .017	.779 ± .017	.477 ± .006	.525 ± .009	.502 ± .007
PCA	.797 ± .006	.782 ± .005	.788 ± .006	.491 ± .001	.561 ± .002	.526 ± .003
PCA-lr	.838 ± .006	.820 ± .004	.827 ± .005	.492 ± .002	.550 ± .001	.528 ± .003
β-VAE, β=1	.748 ± .013	.733 ± .013	.739 ± .013	.436 ± .035	.526 ± .014	.502 ± .016
β-VAE, β=2	.709 ± .010	.695 ± .010	.700 ± .010	.447 ± .016	.503 ± .014	.493 ± .008
β-VAE, β=4	.662 ± .013	.650 ± .012	.654 ± .013	.422 ± .013	.469 ± .018	.452 ± .015

4.2 UMAP visualization of latent space

In all figures, results are shown for β-VAE when $\beta = 1$.

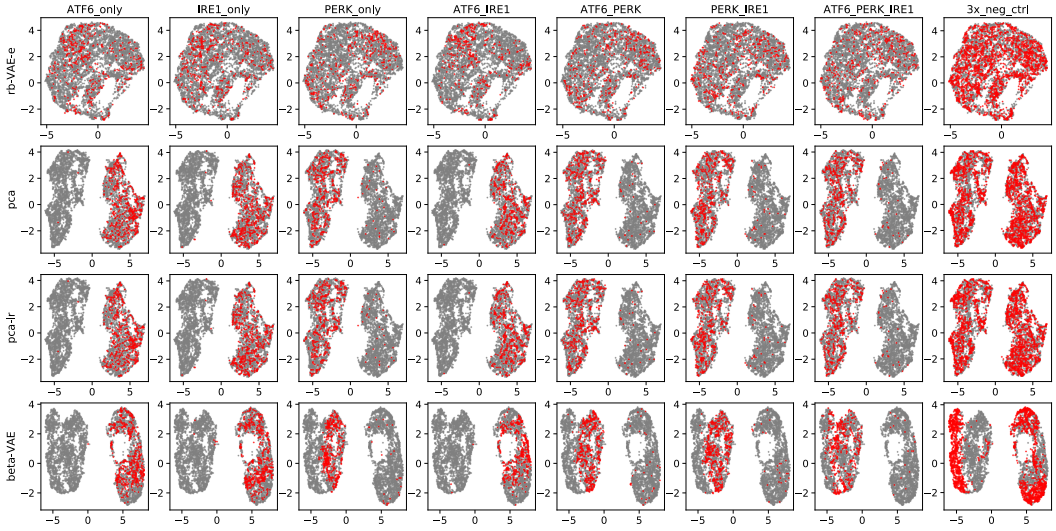

Figure 1: **UMAP visualization of s** for a particular seed. Cells are colored by whether or not they received a particular perturbation.

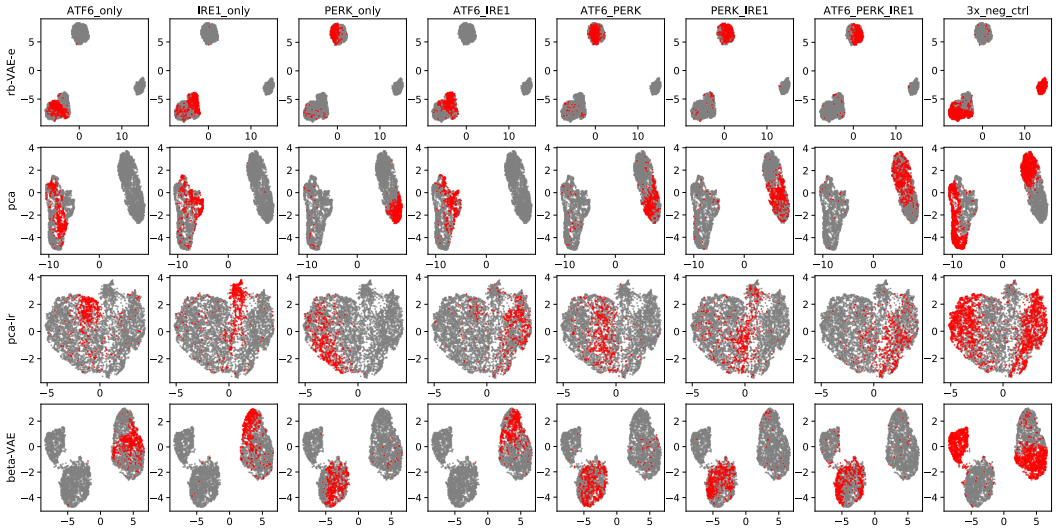

Figure 2: **UMAP visualization of u** for a particular seed. Cells are colored by whether or not they received a particular perturbation.

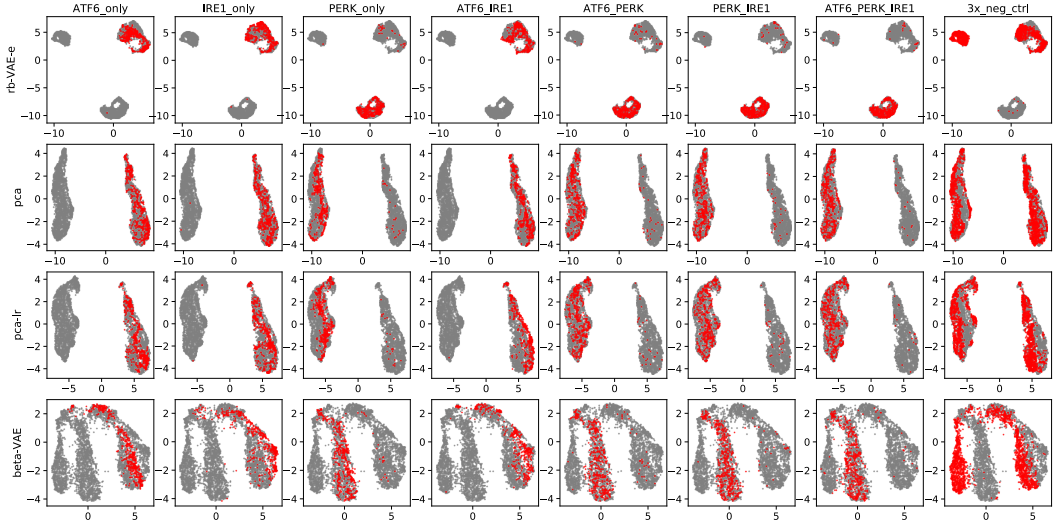

Figure 3: **UMAP visualization of** $[\mathbf{s}; \mathbf{u}]$ for a particular seed. Cells are colored by whether or not they received a particular perturbation.

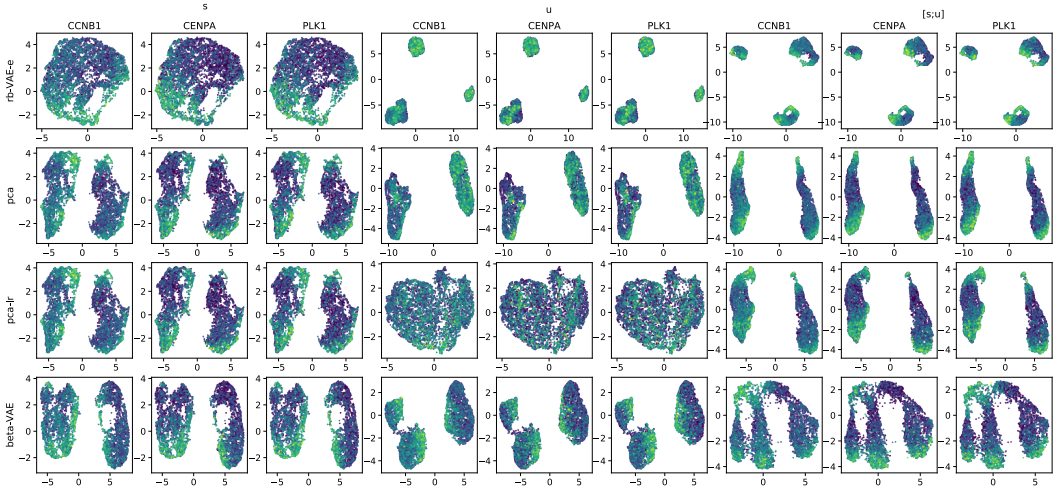

Figure 4: **UMAP visualization of latent variables** for a particular seed. Cells are colored by expression of three cell cycle genes.

References

[1] Adria Ruiz, Oriol Martinez, Xavier Binefa, and Jakob Verbeek. Learning disentangled representations with Reference-Based variational autoencoders. January 2019.

[2] Andrew Butler, Paul Hoffman, Peter Smibert, Efthymia Papalexi, and Rahul Satija. Integrating single-cell transcriptomic data across different conditions, technologies, and species. *Nat. Biotechnol.*, 36(5):411–420, June 2018.

[3] Britt Adamson, Thomas M Norman, Marco Jost, Min Y Cho, James K Nunez, Yuwen Chen, Jacqueline E Villalta, Luke A Gilbert, Max A Horlbeck, Marco Y Hein, Ryan A Pak, Andrew N Gray, Carol A Gross, Atray Dixit, Oren Parnas, Aviv Regev, and Jonathan S Weissman. A multiplexed Single-Cell CRISPR screening platform enables systematic dissection of the unfolded protein response. *Cell*, 167(7):1867–1882.e21, December 2016.

[4] J Pines and T Hunter. Isolation of a human cyclin cDNA: evidence for cyclin mRNA and protein regulation in the cell cycle and for interaction with p34cdc2. *Cell*, 58(5):833–846, September 1989.

[5] David Aristizabal-Corrales, Jinpu Yang, and Fei Li. Cell Cycle-Regulated transcription of CENP-A by the MBF complex ensures optimal level of CENP-A for centromere formation. *Genetics*, 211(3):861–875, March 2019.

[6] R J Lake and W R Jelinek. Cell cycle- and terminal differentiation-associated regulation of the mouse mRNA encoding a conserved mitotic protein kinase. *Mol. Cell. Biol.*, 13(12):7793–7801, December 1993.